



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/712,568	11/13/2003	Zachary Steven Smith	200208663-1	8795

22879 7590 03/26/2007

HEWLETT PACKARD COMPANY
P O BOX 272400, 3404 E. HARMONY ROAD
INTELLECTUAL PROPERTY ADMINISTRATION
FORT COLLINS, CO 80527-2400

EXAMINER

GORTAYO, DANGELINO N

ART UNIT	PAPER NUMBER
----------	--------------

2168

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	03/26/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No. 10/712,568	Applicant(s) SMITH ET AL.	
	Examiner Dangelino N. Gortayo	Art Unit 2168	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 10 October 2006.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-36 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-36 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 13 November 2003 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

Response to Amendment

1. In the amendment filed on 10/10/2006, claims 1, 15, 21, 25, and 28 have been amended. Claims 30-36 have been added. The currently pending claims considered below are Claims 1-36.

Claim Rejections - 35 USC § 101

2. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

3. Claims 1-36 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. For an invention to be statutory, an invention must disclose a "useful, tangible, and concrete result". The claimed invention as a whole must be useful and accomplish a practical application. That is, it must produce a "useful, concrete and tangible result." *State Street*, 149 F.3d at 1373-74, 47 USPQ2d at 1601-02. The purpose of this requirement is to limit patent protection to inventions that possess a certain level of "real world" value, as opposed to subject matter that represents nothing more than an idea or concept, or is simply a starting point for future investigation or research (*Brenner v. Manson*, 383 U.S. 519, 528-36, 148 USPQ 689, 693-96 (1966)); *In re Fisher*, 421 F.3d 1365, 76 USPQ2d 1225 (Fed. Cir. 2005); *In re Ziegler*, 992 F.2d 1197, 1200-03, 26 USPQ2d 1600, 1603-06 (Fed. Cir. 1993)).

Claim 1 recites the limitation "determining an overall statistical probability as to whether the identified bugs are the same as the bug in question" as the last step.

Art Unit: 2168

Claims 15 and 21 also recite a similar limitation. There is no practical application, as determining an overall statistical probability fails to be useful to a user. It is not until the determination step is completed do the claims become more than an abstraction, enabling the functionality to be realized. Since determination by itself is not a tangible result, the claims fail to recite a tangible result as the determining step is not tangible.

Claim 25 recites the limitation "calculate an overall statistical probability of each bug being the same bug as the bug in question using the determined probabilities" as the last step. Claims 28 and 30 also recites a similar limitation. There is no practical application, as calculating an overall statistical probability fails to be useful to a user. It is not until the calculation step is completed do the claims become more than an abstraction, enabling the functionality to be realized. Since determination by itself is not a tangible result, the claims fail to recite a tangible result as the calculating step is not tangible.

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Art Unit: 2168

5. Claims 1-11, 15-17, 20-22, 25-26, 28, and 30-33 are rejected under 35 U.S.C. 103(a) as being anticipated by Glerum et al. ("Glerum" US Patent 6,944,849 B1) in view of Bates et al. ("Bates" US Patent 7,096,458 B2)

As per claim 1, Glerum teaches "A method for identifying similar bugs, comprising:" (see Abstract and column 3 lines 10-33)

"generating a database" (column 3 lines 29-33 and column 8 lines 15-20, wherein a bug database is modified) "that contains database tokens that relate to identified bugs;" (Figures 6A, 6B, column 3 lines 29-33, column 8 lines 41-47, column 9 lines 1-19, wherein BugIDs are identifiers for various bugs, and are contained in asset tags)

"generating input tokens associated with a bug in question;" (Figures 6A, 6B, column 7 lines 61-63, column 8 lines 26-33, 41-47, wherein asset tags, including bug information, are entered into a database)

"scanning the database for occurrences of the input tokens;" (Figures 6A, 6B column 8 lines 48-67, column 9 lines 27-33, wherein various columns in a database table contain information on the number of times an asset, including bug information, occurs in a database)

Glerum does not disclose "and determining an overall statistical probability as to whether the identified bugs are the same as the bug in question."

Bates teaches "and determining an overall statistical probability as to whether the identified bugs are the same as the bug in question." (column 3 lines 35-54, column 8 line 40 – column 9 line 30, wherein an individual weight is assigned to debug entities).

It would have been obvious at the time of the invention for one of ordinary skill in the art to modify Glerum's system for identifying bug information using database entries in the form of asset tags with Bates' method of determining a match score and weight to identify similarity between bugs. This gives the user the ability to compare different bugs using a numerical value. The motivation for doing so would be to better monitor debugging scenarios and to recall debug scenarios across different programs (column 3 line 5-8).

As per claim 2, Glerum teaches "generating a derivative database from a bug database that contains failing results files." (Figure 6A, 6B, column 8 line 44 – column 9 line 44)

As per claim 3, Glerum teaches "generating database tokens from character strings of the failing results files." (Figure 6A, 6B, column 8 line 44 – column 9 line 44, "BugID")

As per claim 4, Glerum teaches "generating tokens for character strings that are proximate to the term "error" in the failing results files." (Figure 6A, 6B, column 8 line 44 – column 9 line 44)

As per claim 5, Glerum teaches "generating tokens for character strings that comprise at least one of letters, numbers, and underscores." (Figure 6A, 6B, column 8 line 44 – column 9 line 44, assert tags and BigIDs)

As per claim 6, Glerum teaches "noting the number of times each token occurs relative to each bug of the database." (Figure 6A, 6B, column 8 line 54 – column 9 line 7, HitsReleased, HitsMaxVerRel, HitsUnreleased)

Art Unit: 2168

As per claim 7, Glerum teaches “generating tokens from character strings of an input failing results file of the bug in question.” (Figure 6A, 6B, column 8 line 44 – column 9 line 44)

As per claim 8, Glerum teaches “scanning the tokens of the database to identify matches for the input tokens.” Figures 6A, 6B column 8 lines 48-67, column 9 lines 27-33,

As per claim 9, Glerum teaches “identifying the number of occurrences of each input token in the database relative to each bug of the database.” (Figure 6A, 6B, column 8 line 54 – column 9 line 7, HitsReleased, HitsMaxVerRel, HitsUnreleased)

As per claim 10, Bates teaches “summing the total number of occurrences of each input token in the database and normalizing the total number of occurrences of each input token as to each bug of the database.” (column 3 lines 35-54, column 10 line 48 – column 11 line 3)

As per claim 11, Glerum teaches “scaling normalized values that result from the normalizing to obtain scaled probabilities as to each input token relative to each bug of the database.” (column 3 lines 35-54, column 11 line 30 – column 12 line 5)

As per claim 15, Glerum teaches “A system for identifying similar bugs, comprising:” (see Abstract and column 3 lines 10-33)

“means for generating input tokens associated with a bug in question;” (Figures 6A, 6B, column 7 lines 61-63, column 8 lines 26-33, 41-47, wherein asset tags, including bug information, are entered into a database)

“means for scanning a database that associates bugs with database tokens pertaining to bugs for occurrences of the input tokens;” (Figures 6A, 6B column 8 lines 48-67, column 9 lines 27-33, wherein various columns in a database table contain information on the number of times an asset, including bug information, occurs in a database)

Glerum does not disclose “and means for determining an overall statistical probability for each bug of the database of being the same bug as the bug in question.”

Bates teaches “and means for determining an overall statistical probability for each bug of the database of being the same bug as the bug in question.” (column 3 lines 35-54, column 8 line 40 – column 9 line 30, wherein an individual weight is assigned to debug entities).

It would have been obvious at the time of the invention for one of ordinary skill in the art to modify Glerum’s system for identifying bug information using database entries in the form of asset tags with Bates’ method of determining a match score and weight to identify similarity between bugs. This gives the user the ability to compare different bugs using a numerical value. The motivation for doing so would be to better monitor debugging scenarios and to recall debug scenarios across different programs (column 3 line 5-8).

As per claim 16, Glerum teaches “means for generating tokens from character strings of an input failing results file for the bug in question.” (Figure 6A, 6B, column 8 line 44 – column 9 line 44)

As per claim 17, Glerum teaches “means for scanning the database tokens to identify matches for the input tokens and means for identifying the number of occurrences of the input tokens in the database relative to each potential bug.” (Figure 6A, 6B, column 8 line 54 – column 9 line 7, HitsReleased, HitsMaxVerRel, HitsUnreleased)

As per claim 20, Glerum teaches “means for generating the database from failing results files contained in a bug database.” (Figure 6A, 6B, column 8 line 44 – column 9 line 44)

As per claim 21, Glerum teaches “A system stored on a computer-readable medium, the system comprising:” (see Abstract and column 3 lines 10-33)

“logic configured to generate a database” (column 3 lines 29-33 and column 8 lines 15-20, wherein a bug database is modified) “that associates bugs with tokens derived from failing results files of the bugs;” (Figures 6A, 6B, column 3 lines 29-33, column 8 lines 41-47, column 9 lines 1-19, wherein BugIDs are identifiers for various bugs, and are contained in asset tags)

“logic configured to generate input tokens from an input that describes a bug in question;” (Figures 6A, 6B, column 7 lines 61-63, column 8 lines 26-33, 41-47, wherein asset tags, including bug information, are entered into a database)

“logic configured to identify the number of occurrences of each of the input tokens in the database as per each potential bug;” (Figures 6A, 6B column 8 lines 48-67, column 9 lines 27-33, wherein various columns in a database table contain

Art Unit: 2168

information on the number of times an asset, including bug information, occurs in a database)

Glerum does not disclose “and logic configured to determine an overall statistical probability of each bug being the same as the bug in question relative to the number of occurrences.”

Bates teaches “and logic configured to determine an overall statistical probability of each bug being the same as the bug in question relative to the number of occurrences.” (column 3 lines 35-54, column 8 line 40 – column 9 line 30, wherein an individual weight is assigned to debug entities).

It would have been obvious at the time of the invention for one of ordinary skill in the art to modify Glerum’s system for identifying bug information using database entries in the form of asset tags with Bates’ method of determining a match score and weight to identify similarity between bugs. This gives the user the ability to compare different bugs using a numerical value. The motivation for doing so would be to better monitor debugging scenarios and to recall debug scenarios across different programs (column 3 line 5-8).

As per claim 22, Glerum teaches “the logic configured to generate input tokens is configured to generate tokens from character strings of an input failing results file.” (Figure 6A, 6B, column 8 line 44 – column 9 line 44)

As per claim 25, Glerum teaches “A bug similarity system stored on a computer-readable medium, the system comprising:” (see Abstract and column 3 lines 10-33)

Art Unit: 2168

“a derivative database generator that is configured to generate a derivative database” (column 3 lines 29-33 and column 8 lines 15-20, wherein a bug database is modified) “that contains a plurality of database tokens that are associated with identified bugs;” (Figures 6A, 6B, column 3 lines 29-33, column 8 lines 41-47, column 9 lines 1-19, wherein BugIDs are identifiers for various bugs, and are contained in asset tags)

“and an similarity calculator that is configured to: generate input tokens from an input that describes a bug in question,” (Figures 6A, 6B, column 7 lines 61-63, column 8 lines 26-33, 41-47, wherein asset tags, including bug information, are entered into a database)

“determine the number of occurrences of the input tokens in the derivative database relative to each bug,” (Figures 6A, 6B column 8 lines 48-67, column 9 lines 27-33, wherein various columns in a database table contain information on the number of times an asset, including bug information, occurs in a database)

Glerum does not disclose “determine the probability of each bug being the same bug as the bug in question relative to each input token, and calculate an overall statistical probability of each bug being the same bug as the bug in question using the determined probabilities.”

Bates teaches “determine the probability of each bug being the same bug as the bug in question relative to each input token, and calculate an overall statistical probability of each bug being the same bug as the bug in question using the determined probabilities.” (column 3 lines 35-54, column 8 line 40 – column 9 line 30, wherein an individual weight is assigned to debug entities).

It would have been obvious at the time of the invention for one of ordinary skill in the art to modify Glerum's system for identifying bug information using database entries in the form of asset tags with Bates' method of determining a match score and weight to identify similarity between bugs. This gives the user the ability to compare different bugs using a numerical value. The motivation for doing so would be to better monitor debugging scenarios and to recall debug scenarios across different programs (column 3 line 5-8).

As per claim 26, Glerum teaches "generate the database tokens from character strings contained in failing results files of a bug database." (Figure 6A, 6B, column 8 line 44 – column 9 line 44)

As per claim 28, Glerum teaches "A computer system, comprising:" (see Abstract and column 3 lines 10-33)

"a processing device;" (Figure 1, reference 20, Figure 2 reference 200)

"and a memory that comprises a bug similarity system, the bug similarity system being configured to generate a first set of tokens for each of several bugs," (column 6 lines 12-40, wherein a client or developer application interacts with a server/database to generate asset tag information, containing bug information, in a database)

"generate input tokens from an input that describes a bug in question," (Figures 6A, 6B, column 7 lines 61-63, column 8 lines 26-33, 41-47, wherein asset tags, including bug information, are entered into a database)

“determine the number of occurrences of the input tokens in the first sets of tokens,” (Figures 6A, 6B column 8 lines 48-67, column 9 lines 27-33, wherein various columns in a database table contain information on the number of times an asset, including bug information, occurs in a database)

Glerum does not disclose “determine the probability as to each of the bugs of whether each bug is the same bug as the bug in question relative to each input token, and calculate an overall statistical probability as to whether the bugs are the same bug as the bug in question using the determined probabilities.”

Bates teaches “determine the probability as to each of the bugs of whether each bug is the same bug as the bug in question relative to each input token, and calculate an overall statistical probability as to whether the bugs are the same bug as the bug in question using the determined probabilities.” (column 3 lines 35-54, column 8 line 40 – column 9 line 30, wherein an individual weight is assigned to debug entities).

It would have been obvious at the time of the invention for one of ordinary skill in the art to modify Glerum’s system for identifying bug information using database entries in the form of asset tags with Bates’ method of determining a match score and weight to identify similarity between bugs. This gives the user the ability to compare different bugs using a numerical value. The motivation for doing so would be to better monitor debugging scenarios and to recall debug scenarios across different programs (column 3 line 5-8).

As per claim 30, Glerum teaches “scanning a bug database to identify all bugs contained in the database and failing results files associated with the bugs;” (column 6 lines 47-52, wherein a program in memory is tested)

“generating tokens that comprise character strings contained in the failing results files;” (Figures 6A, 6B, column 7 lines 61-63, column 8 lines 26-33, 41-47, wherein asset tags, including bug information, are entered into a database)

“creating a bug file for each bug identified in the bug database comprising the tokens that were generated and an indication as to the number of appearances of each token;” (Figures 6A, 6B, column 3 lines 29-33, column 8 lines 41-47, column 9 lines 1-19, wherein BugIDs are identifiers for various bugs, and are contained in asset tags)

“storing the bug files in a derivative database;” (Figure 6A, 6B, column 8 line 44 – column 9 line 44, wherein bug information is stored)

“receiving a failing results file for a bug in question;” (column 7 lines 50-60, wherein an asset will fire after a hit)

generating input tokens that comprise character strings contained in the failing results file of the bug in question;” (Figures 6A, 6B, column 7 lines 61-63, column 8 lines 26-33, 41-47, wherein asset tags, including bug information, are entered into a database)

“searching the bug files of the derivative database to identify occurrences of the input tokens as to each identified bug of the bug database;” (Figures 6A, 6B column 8 lines 48-67, column 9 lines 27-33, wherein various columns in a database table contain

Art Unit: 2168

information on the number of times an asset, including bug information, occurs in a database)

Glerum does not disclose “calculating the statistical probability that the bug in question is one of the identified bug of the bug database.”

Bates teaches “calculating the statistical probability that the bug in question is one of the identified bug of the bug database.” (column 3 lines 35-54, column 8 line 40 – column 9 line 30, wherein an individual weight is assigned to debug entities).

It would have been obvious at the time of the invention for one of ordinary skill in the art to modify Glerum’s system for identifying bug information using database entries in the form of asset tags with Bates’ method of determining a match score and weight to identify similarity between bugs. This gives the user the ability to compare different bugs using a numerical value. The motivation for doing so would be to better monitor debugging scenarios and to recall debug scenarios across different programs (column 3 line 5-8).

As per claim 31, Bates teaches “calculating the statistical probability comprises summing a total number of occurrences as to each token across the derivative database.” (column 3 lines 35-54, column 10 line 48 – column 11 line 3)

As per claim 32, Bates teaches “calculating the statistical probability further comprises normalizing the total number of occurrences of each token relative to each identified bug to generate normalized probabilities.” (column 3 lines 35-54, column 10 line 48 – column 11 line 3)

As per claim 33, Bates teaches “calculating the statistical probability further comprises scaling the normalized probabilities as to each token/bug pair to generate scaled probabilities.” (column 3 lines 35-54, column 11 line 30 – column 12 line 5)

6. Claims 12 and 34 are rejected under 35 U.S.C. 103(a) as being anticipated by Glerum et al. (“Glerum” US Patent 6,944,849 B1) in view of Bates et al. (“Bates” US Patent 7,096,458 B2) and further in view of Jammu et al. (“Jammu” US Patent 6,643,801 B1)

As per claim 12, Glerum and Bates are disclose in claim 11 above. Glerum and Bates does not specifically teach “determining the standard deviance for each scaled probability and removing bug tokens from consideration that are associated with an input token having a deviance below a predetermined minimum deviance.”

Jammu teaches “determining the standard deviance for each scaled probability and removing bug tokens from consideration that are associated with an input token having a deviance below a predetermined minimum deviance.” (Table 1, Table 2, column 7 line 11 – column 8 line 54, wherein a standard deviation of probability is calculated in a fault log, and a sufficiently large number of cases is determined).

It would have been obvious at the time of the invention for one of ordinary skill in the art to modify the system of Glerum and Bates to include the ability to determine the standard deviation of faults in a database and limiting the number of considered cases to a sufficiently large set. This gives the user a more accurate and reliable estimate

Art Unit: 2168

when diagnosing errors. The motivation for doing so would be to identify patterns in fault logs and provide data for matching errors (column 1 line 63 – column 2 line 5).

As per claim 34, Glerum and Bates are disclose in claim 33 above. Glerum and Bates does not specifically teach “calculating the statistical probability further comprises determining a standard deviation of the scaled probabilities.

Jammu teaches “calculating the statistical probability further comprises determining a standard deviation of the scaled probabilities. (Table 1, Table 2, column 7 line 11 – column 8 line 54, wherein a standard deviation of probability is calculated in a fault log, and a sufficiently large number of cases is determined).

It would have been obvious at the time of the invention for one of ordinary skill in the art to modify the system of Glerum and Bates to include the ability to determine the standard deviation of faults in a database and limiting the number of considered cases to a sufficiently large set. This gives the user a more accurate and reliable estimate when diagnosing errors. The motivation for doing so would be to identify patterns in fault logs and provide data for matching errors (column 1 line 63 – column 2 line 5).

7. Claims 13 and 35 are rejected under 35 U.S.C. 103(a) as being anticipated by Glerum et al. (“Glerum” US Patent 6,944,849 B1) in view of Bates et al. (“Bates” US Patent 7,096,458 B2), in view of Jammu et al. (“Jammu” US Patent 6,643,801 B1), and further in view of Chiang et al. (“Chiang” US Patent 7,013,457 B2)

As per claim 13, Glerum, Bates, and Jammu are disclose in claim 12 above.

Glerum, Bates, and Jammu does not specifically teach “determining the overall probability as to all bugs using the scaled probabilities associated with those bugs.”

Chiang teaches “determining the overall probability as to all bugs using the scaled probabilities associated with those bugs.” (column 5 lines 23-41, column 6 lines 21-50, wherein the overall probability of a bug occurring in a program code statement is adjusted with a value of one, which acts as a scaling function that indicates a reduced probability of the statement being the source of a bug)

It would have been obvious at the time of the invention for one of ordinary skill in the art to modify the system of Glerum, Bates, and Jammu to include the ability of Chiang to determine the probability of bugs using a scaling function. This gives the user a reduced probability that the program code statement is the source of the bug. The motivation for doing so would be to reduce the time to debug code, as a user can refer to those lines most likely to contain the source of the bug, as taught by Chiang (column 2 lines 17-22).

As per claim 35, Glerum, Bates, and Jammu are disclose in claim 34 above.

Glerum, Bates, and Jammu does not specifically teach “calculating the statistical probability further comprises determining an overall statistical probability as to each identified bug being the same bug as the bug in question.”

Chiang teaches “calculating the statistical probability further comprises determining an overall statistical probability as to each identified bug being the same bug as the bug in question.” (column 5 lines 23-41, column 6 lines 21-50, wherein the

Art Unit: 2168

overall probability of a bug occurring in a program code statement is adjusted with a value of one, which acts as a scaling function that indicates a reduced probability of the statement being the source of a bug)

It would have been obvious at the time of the invention for one of ordinary skill in the art to modify the system of Glerum, Bates, and Jammu to include the ability of Chiang to determine the probability of bugs using a scaling function. This gives the user a reduced probability that the program code statement is the source of the bug. The motivation for doing so would be to reduce the time to debug code, as a user can refer to those lines most likely to contain the source of the bug, as taught by Chiang (column 2 lines 17-22).

8. Claims 14 and 36 are rejected under 35 U.S.C. 103(a) as being anticipated by Glerum et al. ("Glerum" US Patent 6,944,849 B1) in view of Bates et al. ("Bates" US Patent 7,096,458 B2), in view of Jammu et al. ("Jammu" US Patent 6,643,801 B1), in view of Chiang et al. ("Chiang" US Patent 7,013,457 B2), and further in view of Booth et al. ("Booth" US Patent 5,922,079)

As per claim 14, Glerum, Bates, Jammu, and Chiang are disclose in claim 34 above. Glerum, Bates, Jammu, and Chiang does not specifically teach "applying Bayes' Theorem to the scaled probabilities to calculate the overall probability for each bug as being the same bug as the bug in question."

Booth teaches "applying Bayes' Theorem to the scaled probabilities to calculate the overall probability for each bug as being the same bug as the bug in question."

Art Unit: 2168

(column 4 lines 29-43, column 11 line 51 – column 12 line 35, wherein Bayes' theory is applied in risk estimation when diagnosing faults)

It would have been obvious at the time of the invention for one of ordinary skill in the art to modify the system of Glerum, Bates, Jammu, and Chiang to include Booth's method of assigning weights in fault analysis and diagnosis to calculate the similarities between bugs. This gives the user the ability to assign individual weights to component failures, as in Booth. The motivation for doing so would be to automatically analyze and troubleshoot potential problems with a test suit and modeling errors based on incorrect diagnosis (column 5 lines 34-59).

As per claim 36, Glerum, Bates, Jammu, and Chiang are disclose in claim 35 above. Glerum, Bates, Jammu, and Chiang does not specifically teach "determining an overall statistical probability comprises applying Bayes' Theorem."

Booth teaches "determining an overall statistical probability comprises applying Bayes' Theorem." (column 4 lines 29-43, column 11 line 51 – column 12 line 35, wherein Bayes' theory is applied in risk estimation when diagnosing faults)

It would have been obvious at the time of the invention for one of ordinary skill in the art to modify the system of Glerum, Bates, Jammu, and Chiang to include Booth's method of assigning weights in fault analysis and diagnosis to calculate the similarities between bugs. This gives the user the ability to assign individual weights to component failures, as in Booth. The motivation for doing so would be to automatically analyze and troubleshoot potential problems with a test suit and modeling errors based on incorrect diagnosis (column 5 lines 34-59).

Art Unit: 2168

9. Claims 18 and 23 are rejected under 35 U.S.C. 103(a) as being anticipated by Glerum et al. ("Glerum" US Patent 6,944,849 B1) in view of Bates et al. ("Bates" US Patent 7,096,458 B2) and further in view of Chiang et al. ("Chiang" US Patent 7,013,457 B2)

As per claim 18, Glerum and Bates are disclosed in claim 15 above. Glerum and Bates does not teach "means for determining a probability that a bug is the same relative to each database token associated with the bug."

Chiang teaches "means for determining a probability that a bug is the same relative to each database token associated with the bug." (column 5 lines 23-41, column 6 lines 21-50, wherein the overall probability of a bug occurring in a program code statement is adjusted with a value of one, which acts as a scaling function that indicates a reduced probability of the statement being the source of a bug)

It would have been obvious at the time of the invention for one of ordinary skill in the art to modify the system of Glerum, Bates, and Jammu to include the ability of Chiang to determine the probability of bugs using a scaling function. This gives the user a reduced probability that the program code statement is the source of the bug. The motivation for doing so would be to reduce the time to debug code, as a user can refer to those lines most likely to contain the source of the bug, as taught by Chiang (column 2 lines 17-22).

As per claim 23, Glerum and Bates are disclosed in claim 21 above. Glerum and Bates does not teach "the logic configured to determine the overall probability is

Art Unit: 2168

configured to determine probabilities as to each bug relative to database tokens associated-with those bugs.”

Chiang teaches “the logic configured to determine the overall probability is configured to determine probabilities as to each bug relative to database tokens associated-with those bugs.” (column 5 lines 23-41, column 6 lines 21-50, wherein the overall probability of a bug occurring in a program code statement is adjusted with a value of one, which acts as a scaling function that indicates a reduced probability of the statement being the source of a bug)

It would have been obvious at the time of the invention for one of ordinary skill in the art to modify the system of Glerum, Bates, and Jammu to include the ability of Chiang to determine the probability of bugs using a scaling function. This gives the user a reduced probability that the program code statement is the source of the bug. The motivation for doing so would be to reduce the time to debug code, as a user can refer to those lines most likely to contain the source of the bug, as taught by Chiang (column 2 lines 17-22).

10. Claims 19 and 24 are rejected under 35 U.S.C. 103(a) as being anticipated by Glerum et al. (“Glerum” US Patent 6,944,849 B1) in view of Bates et al. (“Bates” US Patent 7,096,458 B2), in view of Chiang et al. (“Chiang” US Patent 7,013,457 B2), and further in view of Booth et al. (“Booth” US Patent 5,922,079)

As per claim 19, Glerum, Bates, and Chiang are disclosed in claim 18 above. Glerum, Bates, and Chiang does not teach “means for applying Bayes' Theorem to

Art Unit: 2168

those probabilities to calculate the overall probability for each bug as being the bug in question.”

Booth teaches “means for applying Bayes' Theorem to those probabilities to calculate the overall probability for each bug as being the bug in question.” (column 4 lines 29-43, column 11 line 51 – column 12 line 35, wherein Bayes' theory is applied in risk estimation when diagnosing faults)

It would have been obvious at the time of the invention for one of ordinary skill in the art to modify the system of Glerum, Bates, Jammu, and Chiang to include Booth's method of assigning weights in fault analysis and diagnosis to calculate the similarities between bugs. This gives the user the ability to assign individual weights to component failures, as in Booth. The motivation for doing so would be to automatically analyze and troubleshoot potential problems with a test suit and modeling errors based on incorrect diagnosis (column 5 lines 34-59).

As per claim 24, Glerum, Bates, and Chiang are disclosed in claim 23 above. Glerum, Bates, and Chiang does not teach “the logic configured to determine the overall probability is further configured to apply Bayes' Theorem to the determined probabilities to calculate the overall probability for each bug of being the bug in question.”

Booth teaches “the logic configured to determine the overall probability is further configured to apply Bayes' Theorem to the determined probabilities to calculate the overall probability for each bug of being the bug in question.” (column 4 lines 29-43, column 11 line 51 – column 12 line 35, wherein Bayes' theory is applied in risk estimation when diagnosing faults)

Art Unit: 2168

It would have been obvious at the time of the invention for one of ordinary skill in the art to modify the system of Glerum, Bates, Jammu, and Chiang to include Booth's method of assigning weights in fault analysis and diagnosis to calculate the similarities between bugs. This gives the user the ability to assign individual weights to component failures, as in Booth. The motivation for doing so would be to automatically analyze and troubleshoot potential problems with a test suit and modeling errors based on incorrect diagnosis (column 5 lines 34-59).

11. Claims 27 and 29 are rejected under 35 U.S.C. 103(a) as being anticipated by Glerum et al. ("Glerum" US Patent 6,944,849 B1) in view of Bates et al. ("Bates" US Patent 7,096,458 B2), and further in view of Booth et al. ("Booth" US Patent 5,922,079)

As per claim 27, Glerum and Bates are disclosed in claim 25 above. Glerum and Bates does not teach "calculate the overall probability by applying Bayes' Theorem to the determined probabilities."

Booth teaches "calculate the overall probability by applying Bayes' Theorem to the determined probabilities." (column 4 lines 29-43, column 11 line 51 – column 12 line 35, wherein Bayes' theory is applied in risk estimation when diagnosing faults)

It would have been obvious at the time of the invention for one of ordinary skill in the art to modify the system of Glerum, Bates, Jammu, and Chiang to include Booth's method of assigning weights in fault analysis and diagnosis to calculate the similarities between bugs. This gives the user the ability to assign individual weights to component failures, as in Booth. The motivation for doing so would be to automatically analyze and

Art Unit: 2168

troubleshoot potential problems with a test suit and modeling errors based on incorrect diagnosis (column 5 lines 34-59).

As per claim 29, Glerum and Bates are disclosed in claim 28 above. Glerum and Bates does not teach “calculate the overall probability by applying Bayes' Theorem to the determined probabilities.”

Booth teaches “calculate the overall probability by applying Bayes' Theorem to the determined probabilities.” (column 4 lines 29-43, column 11 line 51 – column 12 line 35, wherein Bayes' theory is applied in risk estimation when diagnosing faults)

It would have been obvious at the time of the invention for one of ordinary skill in the art to modify the system of Glerum, Bates, Jammu, and Chiang to include Booth's method of assigning weights in fault analysis and diagnosis to calculate the similarities between bugs. This gives the user the ability to assign individual weights to component failures, as in Booth. The motivation for doing so would be to automatically analyze and troubleshoot potential problems with a test suit and modeling errors based on incorrect diagnosis (column 5 lines 34-59).

Response to Arguments

12. Applicant's arguments, see page 12, filed 10/10/2006, with respect to the 35 USC 101 rejection of claims 1-29 have been fully considered and are not persuasive.

- a. The applicant argues that the claimed inventions do provide a “useful, tangible, and concrete” result, that the probability determination generated by the inventions is useful and tangible, citing sections of the background and

specification of the instant application. Examiner disagrees because although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993). The specification states (page 3 lines 6-14) that "the user can be provided with information that the user can use to make his or her own determination as to whether the bug in question is a new or a previously identified bug", which discloses how overall statistical probability is used. Yet the claims merely state determining or calculating an overall statistical probability as to whether the identified bugs are the same as the bug in question, lacking any sort of step after to indicate how or why statistical probability is used. A determination or calculating step merely calculates the probability, and nothing is done with the probability afterwards, nor is the probability recorded or used anywhere, making the probability not a tangible result. Therefore, the 35 USC 101 rejection of claims 1-29 stand, and a similar rejection to claims 30-36 are made.

13. Applicant's arguments with respect to the 35 USC 102 rejection of claims 1-29 have been considered but are moot in view of the new ground(s) of rejection. The amendments to the independent claims necessitated new grounds of rejection.

Conclusion

14. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP

Art Unit: 2168

§ 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

15. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Dangelino N. Gortayo whose telephone number is (571)272-7204. The examiner can normally be reached on M-F 7:30-4:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tim T. Vo can be reached on (571)272-3642. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Art Unit: 2168

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Dangelino N. Gortayo
Examiner

Tim T. Vo
SPE

DN
3/9/07



TIM VO
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100